AMENDMENTS TO THE SPECIFICATION:

Please replace paragraph [0004] with the following amended paragraph:

[0004]   Installation of a marking machine or other business device is only the first step in the majority of its lifecycle. Most devices are involved in ongoing business processes between the product owners (users), the manufacturer of the product, and/or third party suppliers. Companies that manufacture marking devices typically include products and services in support of users' documents and hope the users will use and live with the offerings for quite a while. This post-sale period presents an opportunity for building a strong and mutually beneficial, long-term relationship between the manufacturer and the users. The post-sale relationship can be defined not only by what the devices do for users, but how they do it, how manufacturers support them, how manufacturers ~~treats~~ treat the users, and how easy it is to own and use the devices overall. Understanding this, embodiments ~~addresses~~ address users' complementary needs to receive services in support of the devices they use: post-sale lifecycles, break-fix needs, and integrated business processes are addressed in various embodiments. These processes range from break-fix service (repairs), to ongoing supply of consumables and supplies, to product upgrades, enhancements, and integration into solutions and other offerings. Traditionally, these post-sale processes were manual in nature and required the device owner/user to play an active role in relaying limited information to manufacturers and suppliers at the time of need.

Please replace paragraph [0007] with the following amended paragraph:

[0007]   Disadvantages of current systems include tight coupling of communication method and system architecture, one-size fits all deployment and integration strategics, and typically no support for devices already deployed. Systems that do offer support for devices already deployed typically are inconsistent between how already deployed devices and new devices are handled. Additionally, systems typically do not include ~~an ability~~ any means for rapid upgrade, extension, customization, and evolution of features, processes, and workflows and are often

limited to basic business processes, failing to provide external services and solutions APIs in a consistent fashion. Generally, and almost across the board, systems treat the device as a simple repository of information, rather than an active participant in the services enabled. Devices must continue to have their mainline feature sets enhanced to stay competitive. In document systems, for example, speeds, feeds, image quality, and document workflows are typically characteristics that are enhanced to render devices competitive. However, increased post-sale interaction between devices, users, and suppliers, and the ability to integrate products into solutions and services and vice versa are becoming points of distinction between devices in the marketplace. In the near future, devices' success and value will likely be measured by the ability of devices to actively participate in their post-sale lifecycles, their ability to seamlessly integrate with solutions offerings, and their capacity for customization and extension based on user needs and requirements. The results of such device abilities are improved ease of use for the user, more effective support from manufacturers, and better overall user satisfaction.

Please replace paragraph [0012] with the following amended paragraph:

[0012]   Services offered to users prior to the instant system were assembled and managed end-to-end within specific product families. This required product teams to invest in developing[[,]] not only the product itself, but also the infrastructure, services, and back-office connections necessary to get the job done. This effort was often very difficult to sustain long-term and was often duplicated across product families.

Please replace paragraph [0017] with the following amended paragraph:

[0017]   An agent software component embedded into devices, add-on modules, and device proxies provides a common device model, common information management (CIM) application programming interface (API), and an environment in which device services can run. A common abstraction of a communication mechanism allows the system to be independent of the physical transport linking nodes. A service model supports services that run close to the device and their

lifecycle, which includes the methods and processes for effective management and customization of services and solutions. As a result, services that are once written to the agent are capable of running on any device, add-on module, or proxy that includes the agent. This yields a system that enables devices and device proxies to be deployed and work together seamlessly from the point of view of the services, as well as policy-based provisioning for device-based services with both user and supplier inputs. The embedded service agent takes an active ~~roll~~ role in solutions offerings and works in coordination with distributed solutions and/or a network-accessible server to provide required functionality. The server provides a clearing house for messages that must traverse the system and provides management functionality necessary to connect and customize distributed services at multiple levels of granularity.

Please replace paragraph [0019] with the following amended paragraph:

[0019]   Embodiments respond to user need and interest by including, for example, a new class of remote services. These services will capitalize on the increased connectivity of devices in the user environment, and utilize embedded computations within the devices themselves to make devices active participants in simplifying user work processes. The platform enables a standards-based solution that can be used to modularly implement remote service offerings in a cross-platform manner that all use [[a]] common back-office integration and work processes. Specific examples of the types of services that can be offered in embodiments include: automated meter reads, automated supplies ordering, productivity reporting, software download, assisted user self-help, remote diagnostics, and prognostics.

Please replace paragraph [0026] with the following amended paragraph:

[0026]   Unifying and managing multiple access needs to several disparate data access mechanisms into one physical entity[[.]] ~~Using~~ using largely COTS PC hardware rather than customized PWBs and making it easy to upgrade over time.

Please replace paragraph [0027] with the following amended paragraph:

[0027]   Architected as a system component suitable for use in any system configuration.[[·]] Providing a platform for continuing assisted-self help service offering extensions over the life of the product family.

Please replace paragraph [0028] with the following amended paragraph:

[0028]   Architecting the CS Platform to enable it to be easily reconfigured for use on other platforms if desired._ Providing device-centric services such as remote monitoring, automated billing, and supplies replenishment (to name a few).

Please replace paragraph [0030] with the following amended paragraph:

FIG. 2 is an another schematic illustration of the overall architecture of embodiments.

Please replace paragraph [0051] with the following amended paragraph:

[0051]   Embodiments provide a system 1 (FIG. 1) composed of several types of distributed software and hardware components that ensure physical and logical system design flexibility and responsibility of the components. Embodiments employ an architecture including, for example, devices 110 in the user/user environment 100, an asset management system 200 that can be in the user's network or environment 100, and a services host 310 that provides services 320 to which devices can subscribe. System management and services are provided in a system where devices are active participants in both their own services and lifecycle needs as well as those services and lifecycles in which they are only a part.

Please replace paragraph [0059] with the following amended paragraph:

[0059]   These components will typically be distributed across the user's environment 100 as well as at the supplier 300. Together, they provide a flexible end-to-end system 1 for connecting products (such as devices 110 and services 140) to post-sale solutions offerings (additional services 140). The system 1, in embodiments, is designed to provide an architecture in support of a series of deployment options in various physical locations and configurations. Preferably, embodiments provide the broadest device coverage and most rapid deployment of capability for machines in the field and new products in such a way that isolates changes at the device 110 from changes at the back-office 300. Embodiments further provide a unique, value-added, agent software component, the DMA 120, embedded into devices 110, add-on modules 115 (as shown in FIGS. 12, 17 and 18), and/or device proxies 210 that provide the common device model 122, DMTF CIM API 123, and new device services environment 124. Additionally, embodiments can provide a common abstraction of the communication mechanism(s) that allows the system to be independent of any physical transport linking the nodes (devices to supplier systems, etc.), providing greater flexibility and deployment customization based on user requirements. The service model of embodiments supports services that run "close to the device" and their lifecycle, which includes methods and processes for effective management and customization of services and solutions. Services in embodiments once written for the DMA 120 can run on any such enabled device 110 or proxy 220, and devices and device proxies can be deployed and work together seamlessly from the point of view of the services. Provisioning in embodiments can be accomplished on a policy basis for device based services based on both user and supplier supplied information, and services can be made available with rapidity.

Please replace paragraph [0060] with the following amended paragraph:

[0060]   The DMA 120 in embodiments takes an active roll role in solutions offerings and works in coordination with the distributed solutions. These distributed device agents 120 work together with a server 310 at the supplier 300 accessible over a network, such as the Internet or a telephone system. The server's role is to provide a clearinghouse for messages that must traverse the solution and to provide management functionality necessary to connect and customize the distributed services at multiple levels of granularity.

Please replace paragraph [0061] with the following amended paragraph:

[0061]   For devices 110 already deployed that do not include this functionality, an option to add a physical system component 115 (as shown in FIGS. 12, 17 and 18) to the device 110, internally or externally, that enables this functionality is provided by embodiments. To the inventive system 1, a device 110 enabled in this fashion will look no different than a device 110 with the capabilities embedded, as long as the add-on component 115 has a rich interface to the device 110. For example, embodiments including such an add-on component 115 can have the component mounted on the input-output terminal (IOT) of a marking machine, connected to the IOT via EPSV, PWS, and potentially CAN Bus interfaces, and connected to a network. This configuration gives the IOT the capability to participate in device services 140. These add-on components 115 can then be found in a one to one mapping with the device because of the need to access non-standard, or non-network accessible APIs and interfaces in order to offer the full range of device capabilities to the DMA and services platform.

Please replace paragraph [0067] with the following amended paragraph:

[0067]   The DMA services preferably can monitor device events and take prescribed actions. The DMA 120 can preferably publish data to subscribers/users upon occurrence of an event of interest and can preferably invoke methods, such as

diagnostic routines, on the device 110 as directed by internal or external clients or users. This moves device specific processing closer to the device 110 from a centralized application server 320. The role of the applications server 320 transforms from a ~~compute~~ computer platform for execution of applications/services to the management and configuration of applications/services 140. Thus, devices 110 become active participants in the process, as opposed to being passive data repositories in strict client/server architectures.

Please replace paragraph [0068] with the following amended paragraph:

[0068]   The DMA 120 according to embodiments can also perform dynamic updates of services 140 and support components operating within the end-to-end DCS platform 1. Devices 110 that employ the DMA 120 can add new service components 140 dynamically. It allows a user or application component already on the device 110 to request such additions to support services 140. It can also allow the addition or deletion of components as needed and without system or DMA recompilation or restart. In embodiments, the target device 110 itself initiates the additions of a new or upgraded service as a whole or as supporting components for existing services. Thus, in the system 1 described herein, the device 110 can now be responsible for initiating the activity to maintain itself and system management services running on it.

Please replace paragraph [0072] with the following amended paragraph:

[0072]   Alternatively, the DMA 120 can be embedded in a specialized hardware device or add-on component 115 to devices 110 that are standalone, such as copiers, or for existing devices in field that are not able to ran the DMA 120. Such add-on components 115 are shown schematically in ~~FIGS. 12, 16, and 17~~ FIGS. 12, 17 and 18, and will be discussed in more detail below.

Please replace paragraph [0092] with the following amended paragraph:

[0091]   Just as multiple paths can enhance deployment flexibility, it is beneficial to make those paths invisible from the standpoint of the services provider. Preferably, embodiments decouple the devices 110 and proxies 220 from the back office systems 310 as much as possible. A strong abstraction and decoupling of these two halves makes it possible to deploy a capability in devices 110 or the back-office 300 in a staged and independent fashion. In addition, if changes need to be made to systems on either end, the changes will not ripple throughout the overall system 1 if proper abstractions are enabled, enhancing maintainability.

Please replace paragraph [0105] with the following amended paragraph:

[0105]   The architecture and implementation of a provisioning server 900 running, for example, in the services host 310 that meets all the requirements in this section is schematically illustrated, for example, in Table 1 and FIG. 20. Working from left to right in FIG. 20, the first major module is the Service Consumer Interface 901. It is preferably responsible for all interactions with External Users and External Devices 110, 220. It also preferably isolates the other PS modules from the different protocols that Devices and Customers may use. The preferred protocol in embodiments is Web Services, but in the future may be extended to http, email, cellular or other transmission formats. For incoming transactions, it routes the transactions to the correct internal resource to process the request. For outgoing transactions, it takes the outputs of other PS modules that have been queued for a Device or User and translates them into the required protocol required to interact wit with the Device or Customer.

Please replace paragraph [0111] with the following amended paragraph:

[0111] Embodiments apply soft computing techniques, such as, for example, rules and constraints, as a general solution to flexibly model, develop, and examine service policy. The provisioning decision itself is less important overall. That is, given a device 110 that needs a service 140, the PS 900 determines whether it is allowed, whether there is a bundle (the collection of code files that make up the service to be

installed) that is compatible with the device 110 operating parameter information (model type, OS version, etc.), which of a plurality of bundles should be selected if there are a plurality, and what the parameter settings (if any) for the service 140 should be. Generally, in embodiments, code ~~can not~~cannot be written that implements "business rules" that can be used to resolve the questions above. Coding would be required for every change of a rule, the rules would not be directly inspectable by policy makers, and it would assume that each question is separable from the others. Further, it assumes that there is a single policy maker that determines the answers for all the above questions. Thus, an alternate solution must be, and is, provided, in embodiments.

Please replace paragraph [0115] with the following amended paragraph:

[0115]   To summarize, the service subscription and deployment method includes identification by a user or user DMA 120 of a service offering 140 of interest and a request for activation of such service (block 501). During a scheduled check in with the edge host, or during a special connection for the purpose, the DMA 120 sends a message for the supplier system 300 regarding the interest and requested activation. The supplier system 300 retrieves the message from the edge host 410 and applies business rule and work processes to determine user eligibility (block 502). If the user is approved, the supplier system 300 ~~notify~~notifies the edge host 410 that the requested service 140 can be added (block 503). The next time the DMA 120 checks in with the edge host 410, it receives the message that the service 140 can be added (block 504). The DMA 120 then activates the service 140, downloading and/or installing it if necessary (block 505). The new service is then deployed and running (block 506).

Please replace paragraph [0141] with the following amended paragraph:

[0141]   Embodiments further enable the rapid addition and roll out of new services to already deployed systems. For example, say that soon after the launch of a new product a new diagnostic service is developed based on lessons learned ~~form~~

10

from the first three months of its operation in the field. The exact nature and behavior of this service could not have been anticipated when the product was launched, so the diagnostic service would not have been included in the launched product. Embodiments allow such a diagnostic service to be added to installed devices at substantially any time.

Please replace the last sentence of paragraph [0144] with the following amended sentence.

[0144]   Another variant in deployment is to fully embed the DMA into the product itself. This implementation is in a way very similar to the Example 1 implementation in that they are both DMA enabled platforms. For this example however, the small footprint DMA services platform is embedded into the product and communicates with both a Print Station Interface Platform (PSIP) and with an embedded device controller. The limited resources required by the small footprint system is acceptable to that product and development and integration of the required interface components is are relatively easy.

Please replace paragraph [0153] with the following amended paragraph:

[0153]   All options are attractive because as a group they can provide additional flexibility for deployments that will meet a variety of user requirements. The preferred method of connecting, when feasible, is Option A - Wired connectivity via LAN and Internet. This is the option of least development investment and least operation expense. In the short-term this is especially important while the value of the services are is being proven and resources need to be focused on initial services development and delivery — not additional ways to connect to devices. It does not, however, address unconnected devices that will initially be left out of the services if only this option is pursued. For the time being each service will need to consider how to manually include non-connected devices in the offerings.

Please replace paragraph [0187] with the following amended paragraph:

11

[0180]   Embodiments contemplate installation of the CS Platform on a network connected personal computer on the same subnet as the CS platform 115. The install process, a schematic illustration of which is shown in FIG. 19, uses a combination of standard networking utilities and LED indications found on the back of the CS Platform to walk the installer through the process. Since the CS Platform 115 is preferably a headless embedded system, the install process can be tricky. The steps listed here are one possible way to do the install, though others are possible. The combination of feedback on the command screen and LEDs on the device provide a robust process for the installation. The component 115 is initially in power-on standby (block 801) and is powered on by the user (block 802). Preferably, a status LED or the like first blinks to indicate that the component 115 is booting, and then becomes steady on when the component 115 is ready (block 803). In embodiments, the user reads the MAC address of the component 115 (block 804), opens a command window on the UI (block 805), and enters a command with the MAC address and other information (block 806). The user can then ping the component 115 (block 807) to test it, then wait for an indication of completion (block 808), such as one or more LEDs in a steady on state. The user then goes to the component's web server 130 via a browser (block 809), logs on as the administrator (block 810), and configures network information as required (block 811) to enable the component 115 to communicate with the edge host 410. The component 115 reboots, during which the IOT should be powered down (block 813). Once both have completed their reboot, installation and setup are complete {block 813).